CreateProcess-01

Close process and thread handles after calling CreateProcess()

Sean Barnum, Cigital, Inc. [vita¹]

Copyright © 2007 Cigital, Inc.

2007-03-20

Part "Original Cigital Coding Rule in XML"

Mime-type: text/xml, size: 5578 bytes

triffic type: text/xim, size: 5576 bytes				
Attack Category	Denial of Service	Denial of Service		
Vulnerability Category	Process management	Process management		
Software Context	Threads and Processes	Threads and Processes		
Location				
Description	After calling CreateProcess(), e and thread handles get closed.	After calling CreateProcess(), ensure that process and thread handles get closed.		
	process. A ProcessInformation and this contains handles for the associated with the new process these process and thread handle system will not be able to clear process exits. The system will to once both the child and parent process.	CreateProcess() and related functions create a new process. A ProcessInformation argument is returned, and this contains handles for the process and thread associated with the new process. Remember to close these process and thread handles; otherwise the system will not be able to clean up after the child process exits. The system will be able to clean up once both the child and parent processes exit, but this may tie up resources longer than necessary.		
APIs	FunctionName Con	nments		
	CreateProcess			
	CreateProcessA AN	SI implementation		
	CreateProcessW Uni	code implementation		
	CreateProcessAsUser			
	CreateProcessWithLogonW			
Method of Attack	not get cleaned up, an attacker	By repeatedly creating processes and threads that do not get cleaned up, an attacker can eventually force the system into resource exhaustion and lead to a Dos.		
Exception Criteria				
Solution Applicability	Solution Solution Applicability Description	Solution Efficacy		
	Whenever creating a new process. Close thread and process handles in process	Effective.		

^{1.} http://buildsecurityin.us-cert.gov/bsi-rules/35-BSI.html (Barnum, Sean)

ID: 719-BSI | Version: 3 | Date: 5/16/08 2:39:17 PM

	information struct once they are no longer needed.	
Signature Details	BOOL CreateProcess(LPCTSTR pApplicationName, LPTSTR pCommandLine,LPSECURITY_ATTRIBUTES pProcessAttributes,LPSECURITY_ATTRIBUTES pThreadAttributes,BOOL bInheritHandles,DWORD dwCreationFlags,LPVOID pEnvironment,LPCTSTR pCurrentDirectory,LPSTARTUPINFO pStartupInfo, LPPROCESS_INFORMATION pProcessInformation);	
	BOOL CreateProcessAsUser(HANDLE hToken,LPCTSTR lpApplicationName,LPTSTR lpCommandLine, LPSECURITY_ATTRIBUTES lpProcessAttributes,LPSECURITY_ATTRIBUTES lpThreadAttributes,BOOL bInheritHandles,DWORD dwCreationFlags,LPVOID lpEnvironment,LPCTSTR lpCurrentDirectory, LPSTARTUPINFO lpStartupInfo,LPPROCESS_INFORMATION lpProcessInformation);	
	BOOL CreateProcessWithLogonW(LPCWSTR lpUsername,LPCWSTR lpDomain,LPCWSTR lpPassword, DWORD dwLogonFlags,LPCWSTR lpApplicationName,LPWSTR lpCommandLine,DWORD wCreationFlags,LPVOID lpEnvironment,LPCWSTR lpCurrentDirectory,LPSTARTUPINFOW lpStartupInfo,LPPROCESS_INFORMATION lpProcessInfo);	
Examples of Incorrect Code	STARTUPINFO si; PROCESS_INFORMATION pi; BOOL fSuccess; fSuccess = CreateProcess(NULL, TEXT("MyProgram.exe"), NULL, NULL, TRUE, 0, NULL, NULL, &si, π); [] /* Handles in "pi" not closed, preventing cleanup when MyProgram.exe exits */	
Examples of Corrected Code	STARTUPINFO si; PROCESS_INFORMATION pi; BOOL fSuccess; fSuccess = CreateProcess(NULL, TEXT("MyProgram.exe"), NULL, NULL, TRUE, 0, NULL, NULL, &si, π); []	

	Languages	• C • C++	
Discriminant Set	Operating System	• Windows	
Recommended Resources		 MSDN Processes and Threads³ MSDN CreateProcess reference⁴ 	
Source Reference	default.asp?url=/libr	http://msdn.microsoft.com/library/ default.asp?url=/library/en-us/dllproc/base/ processes_and_threads.asp ²	
	<pre>if (!CloseHandle(pi.hThread)) { / * handle error */ } /* Now resources can be released when MyProgram.exe exits. */</pre>		
	<pre>/* Use handles if needed */ [] if (!CloseHandle(pi.hProcess)) { /* handle error */ }</pre>		

Cigital, Inc. Copyright

Copyright © Cigital, Inc. 2005-2007. Cigital retains copyrights to this material.

Permission to reproduce this document and to prepare derivative works from this document for internal use is granted, provided the copyright and "No Warranty" statements are included with all reproductions and derivative works.

For information regarding external or commercial use of copyrighted materials owned by Cigital, including information about "Fair Use," contact Cigital at copyright@cigital.com¹.

The Build Security In (BSI) portal is sponsored by the U.S. Department of Homeland Security (DHS), National Cyber Security Division. The Software Engineering Institute (SEI) develops and operates BSI. DHS funding supports the publishing of all site content.

3

^{1.} mailto:copyright@cigital.com